Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (currently amended) A system for debugging microcontroller code comprising:

a microcontroller installed on a test circuit, wherein the microcontroller includes a first memory and a first CPU;

an ICE (in circuit emulator) including a second memory and a second CPU coupled to a computer system, wherein the ICE emulates the microcontroller, the microcontroller and the ICE run the microcontroller code in lock step by executing the same instructions using the same clocking signals; and

an interface for coupling the test circuit and the ICE enabling data transmission between the test circuit and the computer system, the computer system configured to compare a content of the first memory against a content of the second memory to verify said lock step.

2. (previously presented) The system as recited in Claim 1 wherein the microcontroller is installed on a pod.

3. (previously presented) The system as recited in Claim 1 wherein the microcontroller is copied in an FPGA (field programmable gate array) of the ICE.

4. (previously presented) The system as recited in Claim 1 wherein the first memory includes a first static random access memory (SRAM) and the second memory includes a second SRAM.

5. (previously presented) The system as recited in Claim 1 wherein the first memory further includes a first plurality of register values and the second memory further includes a second plurality of register values.

6. (original) The system as recited in Claim 1 wherein the first CPU includes a first program counter and the second CPU includes a second program counter, and wherein lock step execution is maintained by maintaining the first program counter and the second program counter in lock step.

7. (original) The system as recited in Claim 1 wherein a user compares a content of the first memory and the content of a second memory for consistency when execution of the microcontroller code is halted.

8. (previously presented) The system as recited in Claim 1 wherein a user compares a state of the first CPU and a state of the second CPU for consistency when execution of the microcontroller code is halted.

9. (currently amended) A method for debugging microcontroller code comprising:

a) initializing a first memory of an ICE (in circuit emulator) and a second memory of a microcontroller with microcontroller test code;

b) executing the microcontroller test code on the microcontroller and on the ICE in lock step by executing the same instructions using the same clocking signals;

c) verifying lock step execution by comparing content of the first memory and content of the second memory;

d) if lock step execution is not verified, reporting an error and saving an execution history using a trace buffer coupled to the ICE; and

e) if lock step execution is verified, continuing execution of the microcontroller test code.

10. (original) The method of Claim 9 further comprising:

locating an error within the microcontroller test code by tracing the execution history using the trace buffer.

11. (original) The method of Claim 9 further comprising:

verifying lock step execution by comparing register contents of the first memory and register contents of the second memory.

12. (original) The method of Claim 9 wherein the ICE is implemented using an FPGA (field programmable gate array).

13. (original) The method of Claim 9 wherein the microcontroller is a production microcontroller.

14. (original) The method of Claim 9 further comprising:

halting the execution of the microcontroller test code when a breakpoint is encountered; and

verifying lock step execution by comparing content of the first memory and content of the second memory while the execution is halted.

15. (currently amended) A system for maintaining lock step execution of microcontroller test code during a debugging operation comprising:

a microcontroller installed on a test circuit, wherein the microcontroller includes a first memory;

an ICE (in circuit emulator) including a second memory, wherein the ICE emulates the microcontroller, the microcontroller and the ICE configured to run

the microcontroller code in lock step by executing the same instructions using the same clocking signals;

a computer system coupled to the ICE for controlling a debugging operation on the microcontroller code; and

an interface for coupling the test circuit and the ICE for data transmission, the computer system configured to access the ICE and access the test circuit to compare a content of the first memory against a content of the second memory to verify a lock step execution of the microcontroller code during the debugging operation.

16. (previously presented) The system as recited in Claim 15 wherein the microcontroller is installed on a pod.

17. (previously presented) The system as recited in Claim 15 wherein the microcontroller is copied in an FPGA (field programmable gate array) of the ICE.

18. (previously presented) The system as recited in Claim 15 wherein the first memory further includes a first plurality of register values and the second memory further includes a second plurality of register values.

19. (original) The system as recited in Claim 15 wherein a user compares the content of the first memory and the content of a second memory for consistency when execution of the microcontroller code is halted.

20. (original) The system as recited in Claim 15 wherein the microcontroller is a production microcontroller.